

*MITCHELL
7N-53-CR
OCIT
024336*

Semi-Annual Status Report

Models, Training and Aiding

in

Aerospace Systems

**Christine M. Mitchell
Center for Human-Machine Systems
Industrial and Systems Engineering
Georgia Institute of Technology
Atlanta, GA, 30332-0205
404/894-4321
404/894-2301 (fax)
cm@chmsr.gatech.edu**

Georgia Tech Account E24-T10

**NASA Ames
Grant NCC 2-961
Everett A. Palmer, Technical Monitor**

January 31, 1997

Crew Activity Tracking System (T. J. Callantine)

Overview

The Crew Activity Tracking System (CATS) is an architecture for tracking operator activities in complex systems. CATS uses a model of flight crew activities to predict and explain pilot activities; pilot actions that CATS cannot explain are potential errors. CATS was initially implemented to track the activities of Boeing 757/767 pilots using autopilot flight modes to navigate. An empirical evaluation using airline pilots showed CATS was effective in predicting and interpreting pilot activities. A NASA Technical Memorandum documenting CATS is ready for release following internal NASA review (Callantine, Mitchell, and Palmer, 1997).

Recent Research

Recent CATS research has focused on tracking the activities of flight crews performing the “precision descent,” a new descent procedure being developed at NASA Ames Research Center in parallel with future Air Traffic Management (ATM) automation. The new procedure enables pilots to use existing flight deck automation to comply with descent clearances generated by one component of the new ATM automation, the Center-TRACON Automation System (CTAS) Descent Advisor. CATS research has also focused on datalink communication activities, because datalink technology features prominently in the future ATM system. By enhancing CATS to track precision descent activities in a datalink environment, this research explores the use of CATS as a tool for visualizing pilot performance and identifying potential problems with these new tasks. When the new procedures are established, CATS can be used as the source of knowledge about such activities in training or aiding applications.

Enhancements to CATS have proceeded at two levels. First, several general enhancements were made. A Java version of CATS was developed at NASA Ames to support collaboration with Georgia Tech researchers using Java. A new data server component was added to allow CATS to track precision descent activities using data from the NASA Ames Boeing 747-400 simulator. A prototype connection from the data server to the NASA Mini-ACFS simulator was established to allow the use of Mini-ACFS data. A new interface to CATS was also developed (Figure 1). These general enhancements are products of coordinated research efforts at Georgia Tech and NASA Ames.

Other CATS enhancements specifically addressed visualizing precision descent activities in a datalink ATM environment. First, CATS’ model of crew activities was enhanced to represent various Flight Management Computer (FMC) Control and Display Unit (CDU) programming tasks. Both the precision descent and datalink clearance handling rely heavily on each pilot’s proper use of the CDUs. Activities related to handling voice clearances are also represented in the enhanced model, as are activities related to fine-tuning the aircraft’s speed in descent by adjusting throttles or speed brakes manually. Tasks that have been added to the original implementation of CATS’ model of crew activities (as described in Callantine, et al., 1997) are listed in Table 1. In conjunction with these additions, the model was also enhanced as suggested by the empirical evaluation results and discussion in Callantine, et al., 1997 (e.g., subphases were eliminated, and Mode Control Panel (MCP) target value setting tasks are represented as distinct from mode engagement tasks where they may apply to more than one mode).

The task subtree for the “program cruise altitude” task is shown in Figure 2. All of the other CDU programming tasks are represented analogously in the enhanced model. First, the appropriate CDU page must be accessed, then the value to be entered is placed in the CDU scratchpad. Once in the scratchpad, the value is moved to the appropriate place on the CDU page by line selection keystrokes. Finally,

according to standard operating procedure (“SOP-talk”), after verification by both crew members, the FMS flight plan is modified by pushing the CDU EXEC key. The subtasks shown in Figure 2 are currently decomposed into a single action required to accomplish them; however, each subtask may involve other cognitive and verbal actions. At present, such actions are omitted for parsimony.

An important element of the CATS methodology is the “revision process” by which CATS explains actions it did not initially predict. In the original CATS implementation, the revision process was shown important for explaining actions related to using an autoflight mode other than the predicted mode. Recent research has addressed applying the revision process to unexpected CDU actions. To explain such actions, CATS repeatedly checks whether a valid CDU entry of the required type has been made; if so, unexpected page accesses or scratchpad entries are explained as supporting the associated programming task. Another important enhancement to the revision process concerns flight crew coordination in performing precision descent and datalink clearance handling tasks. Research to further enhance the revision process to explain unexpected actions performed by the wrong crew member according to standard cockpit task allocation guidelines is continuing.

Finally, to support procedure visualization and analysis of precision descent data, CATS was enhanced to produce “tracking reports.” A tracking report provides a summary of CATS’ determinations regarding crew activities. When CATS predicts a pilot action, a tracking report is “opened” that records CATS’ expectation and the time it was produced, any values that are expected to be associated with the activity (e.g., for programming a new cruise altitude, the altitude to be programmed is recorded), and the current operational context. If the expected activity is performed, the tracking report is “closed,” and the CATS explanation CATS, time, any associated values, the agent who performed it (e.g., PF.PNF), and the operational context are all recorded. The tracking report may also be closed if the context changes such that the action is no longer expected. CATS also “opens” tracking reports for unexpected actions, “closing” them when the revision process either succeeds or concludes by noting an inability to explain the unexpected action.

Tracking reports, taken together, produce a useful description of the “flow” of activity. Figure 3 shows a of the Tracking Report window of the CATS interface. Arrowheads that point down indicate when a tracking report was opened; arrowheads that point up indicate when it was closed. Green arrows represent predictions; magenta arrows show that the action was successfully predicted and explained. Yellow arrows indicate unexpected actions were detected. Blue arrows mean the revision process explained the action. Gray arrows mean the action was not explained, or that an expected action was not detected before the context changed and the expectation was removed. The time scale at the top of the window in Figure 3 shows elapsed seconds between each determination. Clicking the mouse on a arrowhead in the Tracking Report window pops up a window that shows the information contained in the tracking report.

To increase the usefulness of tracking reports, several issues deserve consideration. For example, tracking reports are currently logged for actions only. Associating tracking reports with higher level activities (e.g., tasks) may provide a more useful summary description. Further research is needed, however, into what information higher level tracking reports should contain.

References

Callantine, T. J., Mitchell, C. M., and Palmer, E. A. (1997). *GT-CATS: Tracking Operator Activities in Complex Systems* (NASA Technical Memorandum DRAFT).

Case-Based Intelligent Tutoring System

Based on discussions with Ames researchers and pilots, the MD-11 was chosen as the platform for implementation of the Case-Based Intelligent Tutoring System (CB-ITS). Pilots express a significant level of problematic interaction with the automation of the MD-11, resulting in a richer environment for extracting the cases of difficulties to be used in the CB-ITS. Based on this decision, a relationship was

developed with the MD-11 program at Delta airlines to explore ways in which Delta could assist with, and benefit from the proposed research. These efforts resulted in access to MD-11 pilots and flight instructors as well as MD-11 ground school and flight training. Hence, one research team member attended the one the Delta training program for pilots transitioning to the MD-11 in order to gain detailed knowledge of the aircraft and its automation. An effort is underway to develop a similar relationship with Federal Express in order to broaden the input to the research. Two papers on related training research for aviation safety and mode management appeared during this time. One summarized the VNAV Tutor development, evaluation, and results (Chappell et al., 1997). The second paper was presented at the annual Human Factors and Ergonomics conference; it included both a summary of the VNAV research and the new research on the case-based tutor (Chappell and Mitchell, 1996)

References

Chappell, A. R., & Mitchell, C. M. (1996). Use of visualization and contextualization in training operators of complex systems. *Proceedings of the Human Factors and Ergonomics Society, 40th Annual Meeting*, Philadelphia, PA, 264-268.

Chappell, A. R., Crowther, E. G., Mitchell, C. M., & Govindaraj, T. (1997). The VNAV Tutor: Addressing a mode awareness difficulty for pilots of glass cockpit aircraft. *IEEE Transactions on System, Man, and Cybernetics*.

OFMspert Modeling Infrastructure

In order to facilitate visualization and modeling, a tree widget for graphical representation of hierarchically decomposed data was designed and implemented. Given a structure of nodes meeting prescribed requirements, this Java-based program displays a tree of these nodes, as depicted in Figure X. In addition to displaying the tree, the program also allows a user to click on the nodes in order to view node information (e.g., node name (change MCP altitude), time posted, parent and/or child nodes, etc.). In addition, the branches of the tree may be pruned or expanded in order to focus attention of specific aspects of the tree. The tree was written in Java to facilitate cross-platform usability, and hence, has been used on PC, Mac, SUN, and SGI platforms. This tree visualization utility has since been incorporated into several efforts including CATS and procedure visualization.

Procedure Visualization

As part of the NASA Ames effort on procedure visualization, several models of the FANS1 datalink clearance procedure were created and evaluated. Based on discussions of known procedures and automation capabilities, models of pilot's activities in interaction with the automation were created. Using the tree drawing utility discussed above, these models were used to visualize the flow of activity in the cockpit in responding to a FANS1 clearance (see Figure X) in order to better understand/predict points of operational difficulty.

Aircraft Simulation Environment and GT Modifications to SSS

As part of developing a generic simulator for research and demonstration of previous research in the context of a B-757/767 cockpit, work continued to port existing Georgia Tech displays from PHIGS running on Sun Workstations to OpenGL, which will run on a variety of platforms. At this time, B-757 displays of the altimeter, vertical speedometer, attitude direction indication (ADI), horizontal situation indicator (HSI) and the mode control panel (MCP) have been implemented in a combination of C++, OpenGL, and GLUT (OpenGL utility toolkit). This port will allow is a necessary first step to allow the VNAV Tutor to run on Silicon Graphics workstation platform as well as the Sun platform on which it was initially implemented. Concurrently, the VNAV Tutor is being modified to use NASA Ames stone soup simulator as its airplane.

After obtaining the Stone Soup Simulator (SSS) software from NASA Ames in December, modifications have been made, so that the new B-757 displays can be driven by the stone soup simulator. The SSS primary flight display module was modified to spawn a PVM (Parallel Virtual Machine) data server process which in turn broadcasts data from the structure pdf.h to the new B-757 displays, i.e., HIS, ADI, altimeter, etc.

Top of climb (TOC), top of descent (TOD), and end of descent EOD information is required by the HSI. The initial version of SSS did not compute this information ; thus modifications to SSS were made to calculate and provide this information to the required displays.

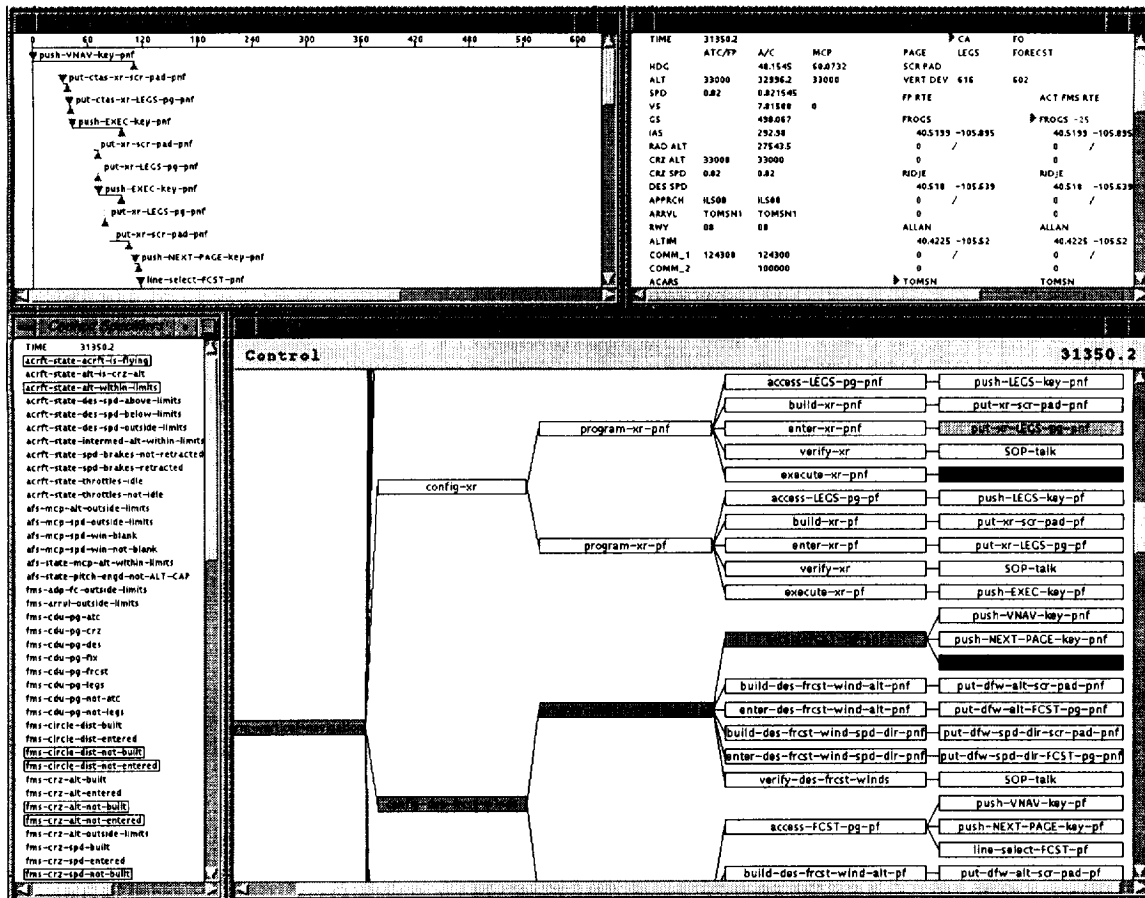
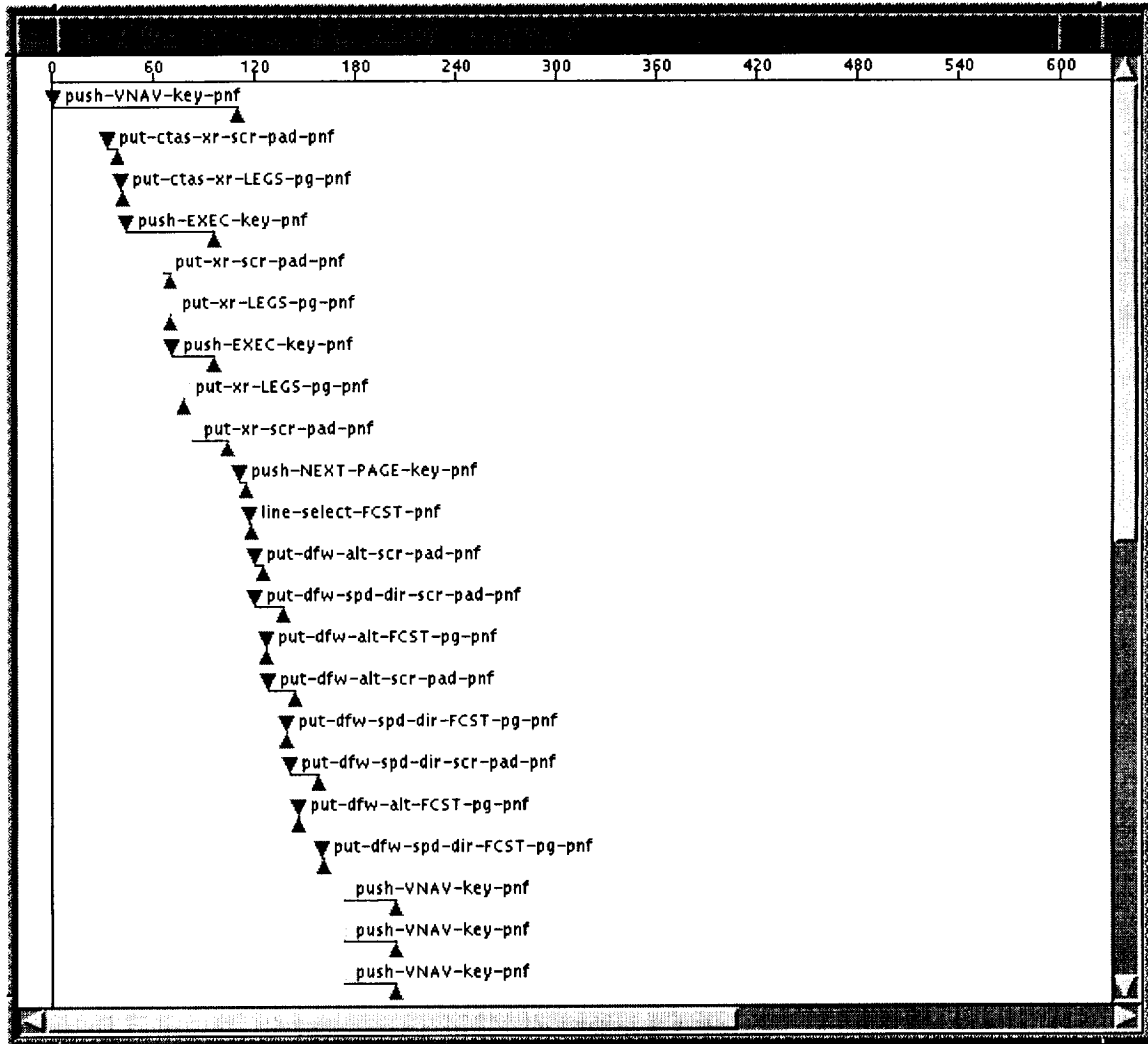


Figure 1. Interface to the Java implementation of CATS.

Table 1. Tasks added to the CATS crew activity model.

1. Program cruise altitude (CDU)
2. Program cruise speed (CDU)
3. Program descent speed (CDU)
4. Program CTAS crossing restriction (CDU)
5. Program CTAS assigned descent point (CDU)
6. Program assigned descent point fix circle (CDU)
7. Program descent forecast winds (CDU)
8. Program route winds (CDU)
9. Program uplinked route (CDU)
10. Program crossing restriction (CDU)
11. Program waypoint (CDU)
12. Program missed approach fix (CDU)
13. Program missed approach crossing restriction (CDU)
14. Program arrival (CDU)
15. Program runway (CDU)
16. Fine-tune speed increase (throttle/speed brake adjustment)
17. Fine-tune speed decrease (throttle/speed brake adjustment)
18. Receive clearance (voice communication)
19. Notify Air Traffic Control (voice communication)
20. Request clearance (voice communication)
21. Request clearance clarification (voice communication)
22. Access FANS clearance (CDU—datalink clearance)
23. Reply to FANS clearance (CDU—datalink clearance)



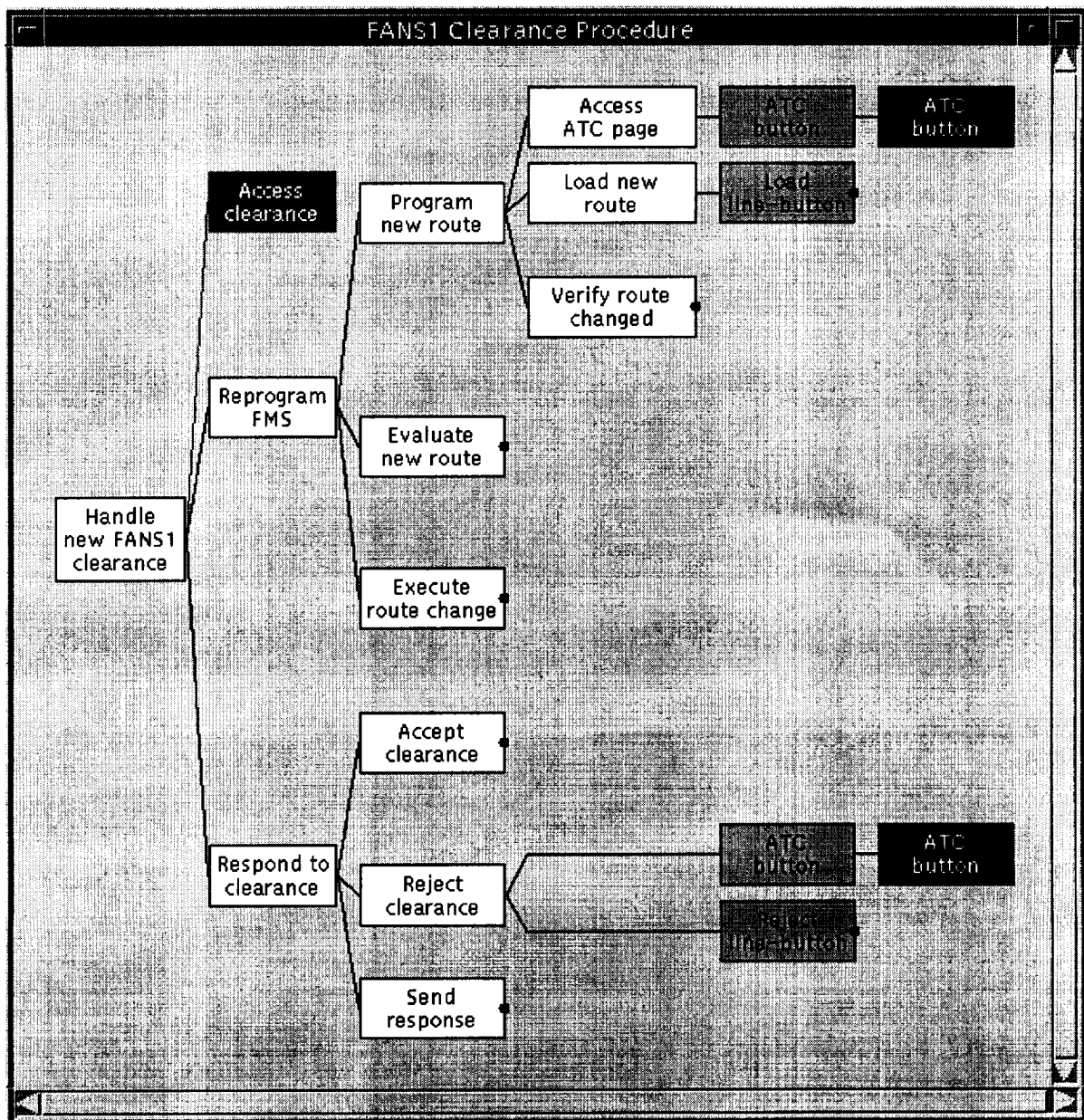


Figure 4. Java Tree Widget depicting FANS-1 Clearance Acceptance Activities.

Appendix

(Copies of Referenced Papers and Reports*)

Callantine, T. J., Mitchell, C. M., and Palmer, E. A. (1997). *GT-CATS: Tracking Operator Activities in Complex Systems* (NASA Technical Memorandum DRAFT).

Chappell, A. R., & Mitchell, C. M. (1996). Use of visualization and contextualization in training operators of complex systems. *Proceedings of the Human Factors and Ergonomics Society, 40th Annual Meeting*, Philadelphia, PA, 264-268.

Chappell, A. R., Crowther, E. G., Mitchell, C. M., & Govindaraj, T. (1997). The VNAV Tutor: Addressing a mode awareness difficulty for pilots of glass cockpit aircraft. *IEEE Transactions on System, Man, and Cybernetics*.

* Please contact Professor Christine M. Mitchell for copies of the reports and papers referenced above.